Docket No. AUS920010564US1

CLAIMS:

What is claimed is:

1.   A method of correcting an error in a processor of a
5   computing system , comprising:
        identifying a failure of a cache array of a
processor;
        deferring deallocation of the processor until reboot
of the computing system;
10   determining, at reboot of the computing system, if
redundancy in the cache array may be used to correct the
failure of the cache array; and
        deallocating the processor, at reboot of the
computing system, if it is determined that redundancy in
15   the cache array cannot be used to correct the failure of
the cache array.

2.   The method of claim 1, further comprising
applying redundancy in the cache array to correct the
20   failure, at reboot of the computing system, if redundancy
in the cache array may be used.

3.   The method of claim 1, wherein identifying a failure
of a cache array of a processor includes determining if a
25   recoverable error threshold is exceeded for the cache
array.

4.   The method of claim 3, further comprising:
        writing threshold exceeded information into
30   persistent memory, wherein deferring deallocation of the
processor includes inhibiting marking the processor as
unavailable.

Docket No. AUS920010564US1

5.   The method of claim 1, wherein deallocating the processor, at reboot of the computing system, includes:
    marking the processor as unavailable if redundancy
5   cannot be used to correct the failure of the cache array.

6.   The method of claim 2, further comprising:
    performing an initial program load after one of applying redundancy in the cache array and deallocating
10 the processor.

7.   The method of claim 5, further comprising:
    determining if a second failure of the cache array occurs at reboot of the computing system;
15   determining if cache array redundancy may be applied to correct the second failure; and
    applying cache array redundancy if cache array redundancy may be applied to correct the second failure.

20 8.   The method of claim 7, further comprising:
    removing cache array status bits and setting one or more bits indicating the processor to be unavailable, if cache array redundancy cannot be applied to correct the second failure.
25

9.   The method of claim 7, wherein applying cache array redundancy includes:
    determining if status bits that are set for the processor are only status bits associated with the cache
30 array; and
    removing status bits for the processor and setting bits indicating that the processor is unavailable, if

Docket No. AUS920010564US1

status bits set for the processor include status bits
other than those associated with the cache array.

10. The method of claim 5, further comprising:

5   determining if another failure of the cache array
occurs at reboot of the computing system; and
        determining if there are status bits set indicating
that the cache array had a predictive failure during
runtime, if another failure did not occur.

10

11. The method of claim 9, further comprising:
        removing the status bits and setting one or more
bits indicating that the processor is unavailable, if
there are status bits indicating that the cache array had

15 a predictive failure during runtime.

12. The method of claim 1, wherein deallocating the
processor includes setting one or more bits indicating
the processor is unavailable.

20

13. An apparatus for correcting an error in a processor
of a computing system, comprising:
        means for identifying a failure of a cache array of
a processor;

25   means for deferring deallocation of the processor
until reboot of the computing system;
        means for determining, at reboot of the computing
system, if redundancy in the cache array may be used to
correct the failure of the cache array; and

30   means for deallocating the processor, at reboot of
the computing system, if it is determined that redundancy
in the cache array cannot be used to correct the failure

Docket No. AUS920010564US1

of the cache array.

14. The apparatus of claim 13, further comprising means for applying redundancy in the cache array to correct the failure, at reboot of the computing system, if redundancy in the cache array may be used.

15. The apparatus of claim 13, wherein the means for identifying a failure of a cache array of a processor includes means for determining if a recoverable error threshold is exceeded for the cache array.

16. The apparatus of claim 15, further comprising:
means for writing threshold exceeded information into persistent memory, wherein the means for deferring deallocation of the processor includes means for inhibiting marking the processor as unavailable.

17. The apparatus of claim 13, wherein the means for deallocating the processor, at reboot of the computing system, includes:
means for marking the processor as unavailable if redundancy cannot be used to correct the failure of the cache array.

18. The apparatus of claim 14, further comprising:
means for performing an initial program load after one of applying redundancy in the cache array and deallocating the processor.

19. The apparatus of claim 17, further comprising:
means for determining if a second failure of the

cache array occurs at reboot of the computing system;

means for determining if cache array redundancy may be applied to correct the second failure; and

means for applying cache array redundancy if cache

5   array redundancy may be applied to correct the second failure.

20.   The apparatus of claim 19, further comprising:

means for removing cache array status bits and

10   setting one or more bits indicating the processor to be unavailable, if cache array redundancy cannot be applied to correct the second failure.

21.   The apparatus of claim 19, wherein the means for

15   applying cache array redundancy includes:

means for determining if status bits that are set for the processor are only status bits associated with the cache array; and

means for removing status bits for the processor and

20   setting bits indicating that the processor is unavailable, if status bits set for the processor include status bits other than those associated with the cache array.

25   22.   The apparatus of claim 17, further comprising:

means for determining if another failure of the cache array occurs at reboot of the computing system; and

means for determining if there are status bits set

30   indicating that the cache array had a predictive failure during runtime, if another failure did not occur.

Docket No. AUS920010564US1

23. The apparatus of claim 21, further comprising:
means for removing the status bits and setting one
or more bits indicating that the processor is
unavailable, if there are status bits indicating that the
5   cache array had a predictive failure during runtime.

24. The apparatus of claim 13, wherein the means for
deallocating the processor includes means for setting one
or more bits indicating the processor is unavailable.
10

25. A computer program product in a computer readable
medium for correcting an error in a processor of a
computing system, comprising:
first instructions for identifying a failure of a
15 cache array of a processor;
second instructions for deferring deallocation of
the processor until reboot of the computing system;
third instructions for determining, at reboot of the
computing system, if redundancy in the cache array may be
20 used to correct the failure of the cache array; and
fourth instructions for deallocating the processor,
at reboot of the computing system, if it is determined
that redundancy in the cache array cannot be used to
correct the failure of the cache array.
25

26. The computer program product of claim 25, further
comprising fifth instructions for applying redundancy in
the cache array to correct the failure, at reboot of the
computing system, if redundancy in the cache array may be
30 used.

27. The computer program product of claim 25, wherein

Docket No. AUS920010564US1

the first instructions for identifying a failure of a
cache array of a processor include instructions for
determining if a recoverable error threshold is exceeded
for the cache array.

5

28.   The computer program product of claim 27, further
comprising:
    fifth instructions for writing threshold exceeded
information into persistent memory, wherein the second
10   instructions for deferring deallocation of the processor
include instructions for inhibiting marking the processor
as unavailable.

29.   The computer program product of claim 25, wherein
15   the fourth instructions for deallocating the processor,
at reboot of the computing system, include:
    instructions for marking the processor as
unavailable if redundancy cannot be used to correct the
failure of the cache array.
20

30.   The computer program product of claim 26, further
comprising:
    sixth instructions for performing an initial program
load after one of applying redundancy in the cache array
25   and deallocating the processor.

31.   The computer program product of claim 29, further
comprising:
    fifth instructions for determining if a second
30   failure of the cache array occurs at reboot of the
computing system;
    sixth instructions for determining if cache array

Docket No. AUS920010564US1

redundancy may be applied to correct the second failure; and

seventh instructions for applying cache array redundancy if cache array redundancy may be applied to

5   correct the second failure.

32.   The computer program product of claim 31, further comprising:

eighth instructions for removing cache array status

10 bits and setting one or more bits indicating the processor to be unavailable, if cache array redundancy cannot be applied to correct the second failure.

33.   The computer program product of claim 31, wherein

15 the seventh instructions for applying cache array redundancy include:

instructions for determining if status bits that are set for the processor are only status bits associated with the cache array; and

20   instructions for removing status bits for the processor and setting bits indicating that the processor is unavailable, if status bits set for the processor include status bits other than those associated with the cache array.

25

34.   The computer program product of claim 29, further comprising:

fifth instructions for determining if another failure of the cache array occurs at reboot of the

30 computing system; and

Sixth instructions for determining if there are status bits set indicating that the cache array had a

Docket No. AUS920010564US1

predictive failure during runtime, if another failure did not occur.

35.   The computer program product of claim 33, further
5   comprising:

instructions for removing the status bits and setting one or more bits indicating that the processor is unavailable, if there are status bits indicating that the cache array had a predictive failure during runtime.

10

36.   The computer program product of claim 25, wherein the fourth instructions for deallocating the processor include instructions for setting one or more bits indicating the processor is unavailable.